Lab 7: Finite State Machine

Date: 12.12.2022

Name: Orkun İbrahim Kök

Section: EE102-01

Purpose

The purpose of this experiment is understanding the logic behind the finite state machines. A finite state machine is asked to be created. The second purpose of the experiment is transferring created finite state machine diagram into the breadboard and showing the states via LEDs on the breadboard.

Design Specifications

I created a finite state machine consisting of:

2 inputs (input1 & input0)

1 output (output)

2 states (1 & 0)

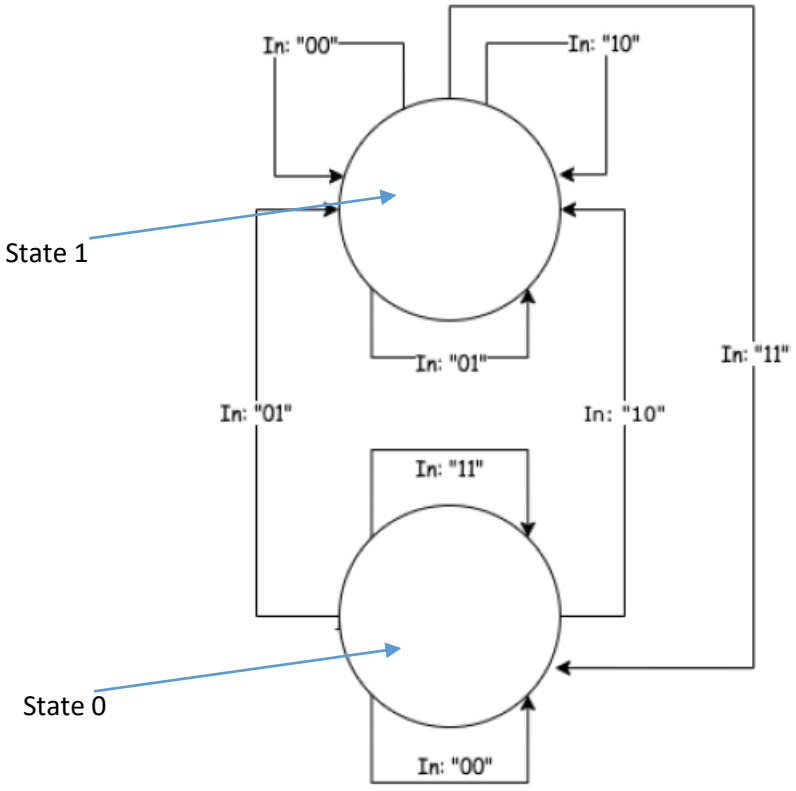

Figure 1: State Diagram

Q (Present State)	INPUT 1	INPUT 0	Q* (Next State)	OUTPUT
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	0
1	0	0	1	1
1	0	1	1	1
1	1	0	1	1
1	1	1	0	0

Figure 2: State Table

The finite state machine (FSM) circuit that I designed includes the following IC's:

- 74 LS/HC 04 (Hex inverter)
- 74 LS/HC 08 (Quad 2-input AND)
- 74 LS/HC 74 (ON Semiconductor - Dual D Flip−Flop)
- 74 LS/HC 32 (Quad 2-input OR gate)
- 74 LS/HC 86 (Quad 2-input XOR gate)

The output of the design only depends on the states therefore the FSM that I designed is a Moore Machine type FSM. The inputs are taken with pushbuttons and the output is displayed by LED on the breadboard.

Methodology

I made the the design and take the necessary logic gates from the lab, which are "NOT, AND, D FF, OR, and XOR gates". Then transferred the circuit to breadboard. The design have 2 inputs, 1 output and 2 states. Inputs are taken by pushbuttons and outputs is shown by LED. After completing the design on breadboard (Figure 3), I tried the input combinations in order to check whether the design is working properly or not. After some trials, everything is working properly.

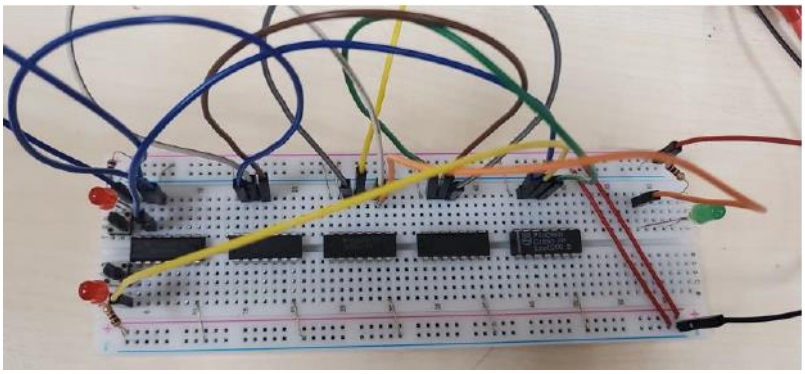

Figure 3: FSM circuit on breadboard

Results

After designing the circuit on breadboard, I tried the input combinations one by one and recorded the outputs and states of the circuit. Figure 4.1-4.5 shows the different input and state combinations.

State = 0

Input1 = 0

Input0 = 0

Output = 0

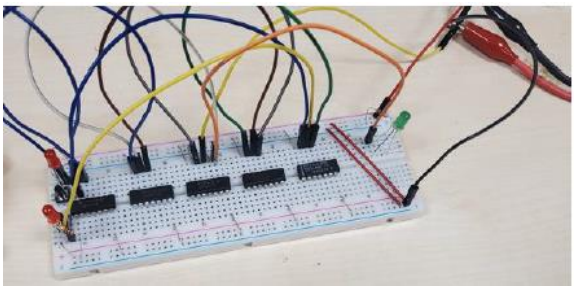

Figure 4.1: 1st Case

State = 1

Input1 = 0

Input0 = 1

Output = 1

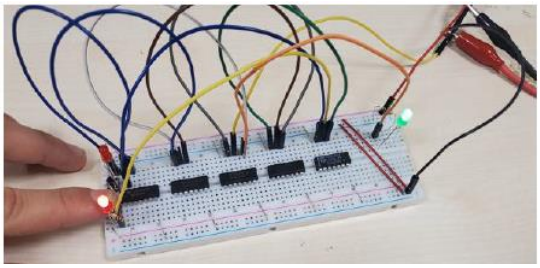

Figure 4.2: 2nd Case

State = 1

Input1 = 1

Input0 = 0

Output = 1

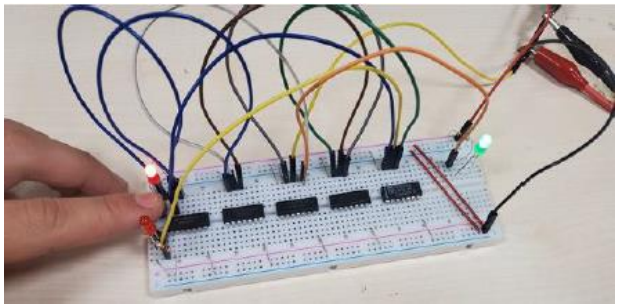

Figure 4.3: 3rd Case

State = 0

Input1 = 1

Input0 = 1

Output = 0

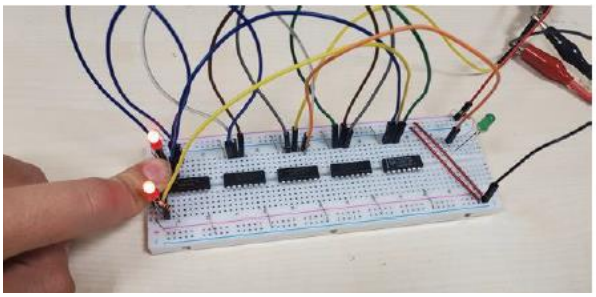

Figure 4.4: 4th Case

State = 1

Input1 = 0

Input0 = 0

Output = 1

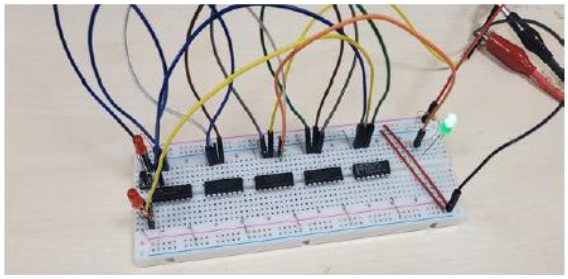

Figure 4.5: 5th Case

Conclusion

The aim of this lab is understanding the logic behind the FSMs and starting from that point, designing a FSM diagram, implementing it to the breadboard with related logic gates. The results at the end are same as the truth table, which is as expected. The designing part of the FSM is a bit challenging because it is taught newly, therefore I needed some time to understand the logic of finite state machines. I choose to use a Moore machine because it is more efficient than Mealy machine, for example less logic gates are used in Moore machine compared to the Mealy machine.